

The embodiments of the invention in which an exclusive property or privilege is claimed are defined as follows:

1. A method for hosting a legacy user interface object originally intended for use in a legacy window manager in a new window manager, comprising:
providing a software bridge between said legacy window manager and said legacy user interface object;
receiving a message at said software bridge intended for said legacy user interface object;
determining whether said message should be forwarded to said new window manager; and
in response to determining that said message should be forwarded, forwarding said message to said legacy window manager.
2. The method of Claim 1, further comprising:
in response to determining that said message should not be forwarded, forwarding said message to a procedure originally intended to handle said message.
3. The method of Claim 2, wherein forwarding said message comprises forwarding said message to a root user interface object hosted in a window tree maintained by said new window manager.
4. The method of Claim 3, further comprising:
routing said message down said window tree maintained by said new window manager to an adapter control associated with said legacy user interface object.
5. The method of Claim 4, further comprising:
processing said message at said adapter control.
6. The method of Claim 5, further comprising:
forwarding said message to said procedure originally intended to handle said message from said adapter control.
7. The method of Claim 6, further comprising:

routing said message from said adapter control to a listener object attached to said adapter control.

8. The method of Claim 7, further comprising:
determining whether said message has been completely handled; and
in response to determining that said message has not been completely handled, routing said message from said adapter control up said window tree maintained by said new window manager so that parent objects of said adapter control may process said message.

9. The method of Claim 8, further comprising:
in response to determining that said message has been completely handled, returning control to a procedure associated with said legacy user interface object.

10. A method for hosting a legacy user interface object originally intended for use in a legacy window manager in a new window manager, comprising:
providing a software bridge between said legacy window manager and said legacy user interface object;
receiving a message at said software bridge intended for said legacy user interface object;
determining whether said message should be forwarded to said new window manager;
in response to determining that said message should be forwarded, forwarding said message to a root user interface object hosted in a window tree maintained by said new window manager;
routing said message from said root user interface object down said window tree to an adapter control associated with said legacy user interface object; and
processing said message at said adapter control.

11. The method of Claim 10, further comprising:
forwarding said message from said adapter control to a procedure originally intended to handle said message.

12. The method of Claim 11, further comprising:

routing said message from said adapter control to a listener object attached to said adapter control.

13. The method of Claim 12, further comprising:
determining whether said message has been completely handled; and
in response to determining that said message has not been completely handled,
routing said message from said adapter control up said window tree maintained by said
new window manager so that parent objects of said adapter control in said window tree
may process said message.

14. The method of Claim 13, further comprising:
in response to determining that said message has been completely handled,
returning control to a procedure associated with said legacy user interface object.

15. A computer-controlled apparatus capable of performing the method of
any one of Claims 1-14.

16. A computer-readable medium comprising instructions which, when
executed by a computer, cause the computer to perform the method of any one of
Claims 1-14.